

AMENDMENTS TO THE CLAIMS

Please amend the claims as follows:

1. (Currently Amended) A method for implicit, unsupervised computation using a ~~digital implementation of a~~ neural network architecture in a computing device wherein the implicit computation comprises local minimilization steps applied to data representations from external input channels or internal connections, said neural network architecture compris[[ing]]es the steps of:

organizing a plurality of computational nodes into a neural computing device, each of the plurality of computational nodes being implemented as a software process on a general purpose computer or by a digital or analog hardware device, wherein the architecture of the neural computing device comprises:

using a plurality of layers, each layer including a subset of the plurality of computational nodes, at least one input processing layer where ~~said the~~ plurality of computational nodes provide a stable transformable digital representation of an input signal, at least one central processing layer, and at least one output processing layer;

using at least one feedforward channel for inputs;

~~processing said data~~ using full lateral and feedback connections within the at least one processing layers;

using full lateral and feedback connections within the ~~said~~ at least one processing layers

using at least one output channel for outputs;

using re-entrant feedback from ~~said the~~ at least one output channel to at least one of the ~~said~~ at least one processing layers;

using local update processes to update each of the plurality of computational nodes where
the such that said transformable digital representations of input signals are distributed to ~~said~~ the
plurality of computational nodes which combine ~~said~~ the stable transformable digital
representations of input signals according to the interconnectivity of ~~said~~ the at least one input
processing layer and ~~said~~ the plurality of computational layers, and which perform
minimalization steps on ~~said~~ the combinations to meet at least one specified success criteria
whereby the local minimalization step receives coincident representations from all afferent links,
selects an output that minimizes the difference between the state represented by said coincident
representations and the at least one specified success criteria, wherein the weight for said
selection is reduced when subsequent coincident representations diverge from said at least one
success criteria, and the said weight for said selection is increased when subsequent coincident
representations converge; and,

outputting from said at least one output processing layer a final digital representation of
the results of said combination and minimization steps to an external device or user.

2. (Original) The method of claim 1, wherein the output channel uses feedforward
connections between the output channel and at least one of the processing layers.

3. (Original) The method of claim 1, wherein the output channel uses bi-directional
connections between the output channel and at least one of the processing layers.

4. (Original) The method of claim 1, wherein the re-entrant feedback is uni-
directional.

5. (Original) The method of claim 1, wherein the re-entrant feedback is bi-
directional.

6. (Previously Presented) The method of claim 1, wherein the local update processes

are any one of: non-stationary random processes, Polya processes and Bose-Einstein processes.

7. (Original) The method of claim 1, wherein the local update processes effect a phase change.

8. (Original) The method of claim 1, wherein the local update processes are equivalent to Bose-Einstein condensation.

9. (Original) The method of claim 1, wherein the local update processes are equivalent to taking a quantum measurement.

10. (Original) The method of claim 1, wherein the local update processes are derived from nearest-neighbor normalization.

11. (Original) The method of claim 1, wherein the local update processes create a Delaunay tessellation from one layer to a next layer.

12. (Original) The method of claim 1, wherein the local update processes include inhibition.

13. (Original) The method of claim 1, wherein the local update processes cause fractal percolation.

14. (Original) The method of claim 1, wherein the minimalization recalibrates the computation nodes in at least one processing layer.

15. (Original) The method of claim 1, wherein the minimalization is triggered by fractal percolation.

16. (Original) The method of claim 1, wherein the minimalization is a quantum measurement.

17. (Original) The method of claim 1, wherein the plurality of layers comprise one module in an architecture that has a plurality of modules.

18. (Original) The method of claim 17, wherein one of the plurality of modules is an attention module with at least two layers connected by bi-directional connections, and with lateral connectivity to at least two processing layers in other modules.

19. (Original) The method of claim 17, wherein one of the plurality of modules is a digital memory.

20. (Original) The method of claim 17, wherein one of the plurality of modules is a dynamic memory having the plurality of layers including the plurality of computational nodes, the computational nodes using feedforward inputs from an output channel of the remainder of the plurality of modules and using feedforward connectivity from a first to a last of the plurality of layers, and

the computational nodes using at least one feedforward re-entrant connection to the first of the plurality of layers and using bi-directional connectivity between the one of the plurality of modules and at least one processing layer from the remainder of modules.

21. (Original) The method of claim 17, wherein the plurality of modules have standardized dimensions, number of layers and number of computational nodes, wherein the computational nodes use connectivity from the input processing layer in each module to neighboring modules, and lateral connectivity between their corresponding processing layers to permit leaky processing.

22. (Original) The method of claim 17, wherein the plurality of modules provide converging and diverging connections from the output layer of one module to at least one processing layer of other modules.

23. (Original) The method of claim 17, wherein each of the plurality of modules includes a logic gate.

24. (Original) The method of claim 17, wherein at least one of the plurality of modules includes a NAND gate.

25. (Original) The method of claim 17, wherein at least one of the plurality of modules includes a CNOT gate.

26. (Original) The method of claim 1, further including the steps of:
copying values for critical probabilities for a lattice formed by the computational nodes from the neural network architecture; and
writing the values to another neural network.

27 (Currently Amended) ~~An data-processing~~ apparatus for implicit digital computation comprising:

~~digital implementation of~~ a neural network architecture means having a plurality of layer means, each layer means ~~including~~ comprising a plurality of ~~adaptive computational node means~~ computational nodes, each of the plurality of computational nodes being implemented as a software process on a general purpose computer or by a digital or analog hardware device, the plurality of layer means comprising; ~~further including~~:

a processing layer means including:
at least one input processing layer means which is capable of rendering a stable transformable digital representation of input signals,
at least one central processing layer means, and
at least one output processing layer means;
feedforward input channel means;
full lateral and feedback connection means within the processing layer means;
output channel means;

re-entrant feedback means from the output channel means to the processing layer means;
means for updating each of the plurality of ~~adaptive~~ computational node means using
local update processes; and

means for using re-entrant feedback from the output channel means to perform
minimalization for general computation such that said stable transformable digital
representations of input signals are distributed to ~~said the~~ plurality of computational nodes which
combine ~~said the~~ stable transformable digital representations of input signals according to the
interconnectivity of ~~said the~~ at least one input processing layer and ~~said the~~ plurality of
computational layers, and which perform minimization steps on ~~said combinations to meet~~
~~specified success criteria whereby the lowest values of said combinations meeting said success~~
~~criteria are retained~~ on the plurality of combinations to match at least one specified success
criteria whereby the said minimalization step selects from an inventory of at least one output
value based on the said plurality of combinations at the moment of the selection step to minimize
the difference between the said plurality of stable transformable combinations and the said at
least one success criteria, whereby the selection weight for said selection is decreased when the
plurality of subsequent input stable transformable combinations diverges from the said at least
one success criteria, and the selection weight for said selection is increased when the plurality of
subsequent input stable transformable combinations converges with said at least one success
criteria.

28. (Original) The apparatus of claim 27, wherein the output channel means uses
feedforward connection means between the output channel means and the processing layer
means.

29. (Original) The apparatus of claim 27, wherein the output channel means uses bi-

directional connection means between the output channel means and the processing layer means.

30. (Original) The apparatus of claim 27, wherein the re-entrant feedback means is uni-directional.

31. (Original) The apparatus of claim 27, wherein the re-entrant feedback means is bi-directional.

32. (Currently Amended) The apparatus of claim 27, wherein the local update processes are any one of: random processes between the plurality of ~~adaptive~~ computational nodes ~~means~~, non-stationary random processes between the plurality of ~~adaptive~~ computational nodes ~~means~~, Polya processes between the plurality of adaptive computational node means, and Bose-Einstein processes between the plurality of ~~adaptive~~ computational nodes ~~means~~.

33. (Currently Amended) The apparatus of claim 27, wherein the local update processes are Bose-Einstein processes and the plurality of ~~adaptive~~ computational nodes ~~means~~ are lasers.

34. (Currently Amended) The apparatus of claim 27, wherein the local update processes are a phase change in the plurality of ~~adaptive~~ computational nodes ~~means~~.

35. (Original) The apparatus of claim 27, wherein the local update processes are a Bose-Einstein condensation in the plurality of ~~adaptive~~ computational nodes ~~means~~.

36. (Original) The apparatus of claim 27, wherein the local update processes are quantum measurements performed on the plurality of ~~adaptive~~ computational nodes ~~means~~.

37. (Original) The apparatus of claim 27, wherein the local update processes perform nearest-neighbor normalization among the plurality of ~~adaptive~~ computational nodes ~~means~~.

38. (Original) The apparatus of claim 27, wherein the local update processes create a Delaunay tessellation from one layer means to a next layer means.

39. (Currently Amended) The apparatus of claim 27, wherein the local update processes include inhibition between at least one of adaptive computational node means and at least one other of the plurality of ~~adaptive computational nodes means~~.

40. (Currently Amended) The apparatus of claim 27, wherein the local update processes cause fractal percolation among the plurality of ~~adaptive computational nodes means~~.

41. (Original) The apparatus of claim 27, wherein the minimalization recalibrates the adaptive computation node means in the processing layer means.

42. (Currently Amended) The apparatus of claim 27, wherein the minimalization step is triggered by fractal percolation across the plurality of ~~adaptive computational nodes means~~.

43. (Currently Amended) The apparatus of claim 27, wherein the minimalization step is a quantum measurement performed on the plurality of ~~adaptive computational nodes means~~.

44. (Original) The apparatus of claim 27, wherein the plurality of layer means is one module means in an architecture with a plurality of module means.

45. (Original) The apparatus of claim 44, wherein one of the plurality of module means is an attention module means that includes:

at least two layer means connected by bi-directional connection means; and

lateral connectivity means to at least two processing layer means belonging to other module means.

46. (Original) The apparatus of claim 44, wherein one of the plurality of module means is a standard digital memory means.

47. (Currently Amended) The apparatus of claim 44, wherein one of the plurality of module means is a dynamic memory means that includes:

a plurality of layer means including a plurality of ~~adaptive computational nodes means~~;

feedforward input means from the output channel means of other module means;

feedforward connectivity means from a first to a last layer means; and

feedforward re-entrant connection means to the first layer means using bi-directional connectivity means between the module means and the processing layer means from the other module means.

48. (Original) The apparatus of claim 44, wherein the module means have standardized dimensions number of layer means and a standardized number of adaptive computational node means, and wherein each module means includes:

connectivity means from the input processing layer means of each module means to a neighboring module means; and

lateral connectivity means between a corresponding processing layer means to permit leaky processing.

49. (Original) The apparatus of claim 44, wherein the plurality of module means provide converging and diverging connection means from the output layer means of each module means to processing layer means of other module means.

50. (Original) The apparatus of claim 44, wherein one module means includes at least one of a logic gate, a NAND gate, and a CNOT gate.

51. (Currently Amended) A method for computation using a ~~digital implementation~~ of a neural network architecture in a computing device comprising the steps of:

organizing a plurality of computational nodes into a neural computing device, each of the plurality of computational nodes being implemented as a software process on a general purpose computer or by a digital or analog hardware device, wherein the architecture of the neural computing device comprises:

using at least one stable transformable digital representation as an input to receive data to be processed from an environment;

using a ~~plurality of~~ locally connected subset of the plurality of computational nodes for fractal percolation using ~~said~~ the at least one stable transformable digital representation[[s]];

using a minimalization step for ~~performing implicit computation upon said data~~; and

using at least one output to output processed data that can be used by a human or as an input to a machine.

52. (Original) The method of claim 51, further including the steps of:

copying values for critical probabilities for a lattice formed by the computational nodes from the neural network architecture; and

writing the values to another neural network.

53. (Original) The method of claim 51, wherein the plurality of locally connected computation nodes has a state space of dimension of at least two.

54. (Original) The method of claim 51, wherein the plurality of locally connected computation nodes has a Hilbert state space.

55. (Original) The method of claim 51, wherein connections among the plurality of locally connected computation nodes extend beyond nearest neighbors.

56. (Original) The method of claim 51, wherein connections among the plurality of locally connected computation nodes are feedforward connections, leading to “first-pass” percolation.

57. (Original) The method of claim 51, wherein at least one connection among the plurality of locally connected computation nodes is an inhibitory connection.

58. (Original) The method of claim 51, wherein the minimalization step is performed

by re-entrant connections to the plurality of locally connected computation nodes.

59. (Original) The method of claim 51, wherein the fractal percolation across the plurality of locally connected computational nodes occurs by one of a random process, a Poisson process, a non stationary random process, a Polya process, and a Bose-Einstein statistical process.

60. (Original) The method of claim 51, wherein the fractal percolation across the plurality of locally connected computation nodes occurs by nearest-neighbor renormalization.

61. (Original) The method of claim 51, wherein the minimalization step is performed by re-scaled weights among the plurality of locally connected computation nodes.

62. (Original) The method of claim 51, wherein the minimalization step is performed by a quantum measurement.

63. (Original) The method of claim 51, wherein the plurality of locally connected computational nodes are one module in an architecture with a plurality of modules.

64. (Original) The method of claim 63 wherein one of the plurality of modules is an attention module using bi-directional connections with at least one other module.

65. (Original) The method of claim 63, wherein one of the plurality of modules is a standard digital memory.

66. (Original) The method of claim 63, wherein one of the plurality of modules is a dynamic memory including:

a plurality of locally connected computational nodes, with feedforward inputs from another module,

feedforward connectivity among the plurality of locally connected computational nodes using at least one feedforward re-entrant connection to the first of the plurality of locally

connected computational nodes; and

bi-directional connectivity between the modules.

67. (Original) The method of claim 63, wherein the plurality of modules have standardized dimensions and numbers of the plurality of locally connected computational nodes, wherein random connectivity among neighboring modules permits leaky processing.

68. (Original) The method of claim 63, wherein the plurality of modules provide converging and diverging connections from the at least one output of one module to another module.

69. (Original) The method of claim 63, wherein each module includes at least one of a logic gate, a NAND gate, and a CNOT gate.

70. (Currently Amended) ~~An data-processing~~ apparatus for implicit ~~digital~~ computation comprising:

~~a digital implementation of~~ a neural network architecture means including:

an input means from an environment capable of rendering a stable transformable digital representation of the environment;

an output means ~~for outputting processed data that can be used by a human or as an input to a machine~~; and,

a plurality of locally connected computation nodes, each of the plurality of computation nodes being implemented as a software process on a general purpose computer or by a digital or analog hardware device, wherein the plurality of locally connected computation nodes is organized to perform ~~the means for~~ fractal percolation using said stable transformable digital representations, wherein a minimalization step is used for computation.

71. (Currently Amended) The apparatus of claim 70, wherein the plurality of locally

connected computation nodes ~~means~~ has a state space of dimension of at least or equal to two.

72. (Currently Amended) The apparatus of claim 70, wherein the plurality of locally connected computation nodes ~~means~~ has a Hilbert state space.

73. (Currently Amended) The apparatus of claim 70, wherein the plurality of locally connected computation nodes ~~means~~ include Rydberg atoms.

74. (Currently Amended) The apparatus of claim 70, wherein the plurality of locally connected computation nodes ~~means~~ include molecular magnets.

75. (Currently Amended) The apparatus of claim 70, wherein the local connection means among the plurality of locally connected computation nodes ~~means~~ extend beyond nearest neighbors.

76. (Currently Amended) The apparatus of claim 70, wherein connection means connecting plurality of locally connected computation nodes ~~means~~ are all feedforward, leading to “first-pass” percolation.

77. (Currently Amended) The apparatus of claim 70, wherein at least one connection means connecting the plurality of locally connected computation nodes ~~means~~ is an inhibitory connection means.

78. (Currently Amended) The apparatus of claim 70, wherein the fractal percolation across the plurality of locally connected computation nodes ~~means~~ occurs by any one of a Polya process, Bose-Einstein condensation, a Poisson process, a non-stationary random process, nearest-neighbor renormalization, and a random process.

79. (Currently Amended) The apparatus of claim 70, wherein the fractal percolation across the plurality of locally connected computation nodes ~~means~~ occurs across an Ising lattice.

80. (Currently Amended) The apparatus of claim 70, wherein the plurality of locally

connected computation nodes ~~means~~ includes at least one laser.

81. (Currently Amended) The apparatus of claim 70, wherein the minimalization step user re-scaled weights among the plurality of locally connected computation nodes ~~means~~.

82. (Currently Amended) The apparatus of claim 70, wherein the minimalization step is performed by one of electron spin resonance pulses, quantum measurement, re-entrant connection means to the plurality of locally connected computation nodes ~~means~~ and coherent radiation.

83. (Currently Amended) The apparatus of claim 70, wherein the plurality of locally connected computational nodes ~~means~~ is one module means in an architecture with a plurality of module means.

84. (Original) The apparatus of claim 83, wherein one of the plurality of module means is an attention module means using bi-directional connection means to connect with another module means.

85. (Original) The apparatus of claim 83, wherein one of the plurality of module means is a standard digital memory means.

86. (Currently Amended) The apparatus of claim 83, wherein one of the plurality of modules is a dynamic memory using a plurality of locally connected computational nodes ~~means~~, and further including:

feedforward input means from the output means of a remainder of the module means;

feedforward connectivity means among the plurality of locally connected computational nodes ~~means~~;

at least one feedforward re-entrant connection means to a first of the plurality of locally connected computational nodes ~~means~~,

bi-directional connectivity means between the module means and both the plurality of locally connected computational nodes ~~means~~ and the remainder of the plurality of module means.

87. (Currently Amended) The apparatus of claim 83, wherein the plurality of module means have standardized dimensions and standardized numbers of the plurality of locally connected computational nodes ~~means~~,

the computational nodes ~~means~~ using random connectivity means to connect to neighboring members of the plurality of module means to permit leaky processing.

88. (Original) The apparatus of claim 83, wherein the plurality of module means provide converging and diverging connection means from one output means of the plurality of module means to at least one other module means.

89. (Original) The apparatus of claim 83, wherein each module means includes one of a logic gate, a NAND gate, and a CNOT gate.

90. (Original) The apparatus of claim 70, further comprising:
storage means for copying values for critical probabilities for a lattice formed by the computational nodes from the neural network architecture; and
transfer means to write the values to another neural network means.

91. (New) A system comprising:
a plurality of computational nodes, wherein each computational node is implemented as a software process on a general purpose computer or by a digital or analog hardware device, each computational node comprising a local update process which transforms data received by the computational node wherein,

a first subset of the plurality of computational nodes is organized into at least one feedforward input channel operatively connected to an input digital or analog data source;

a second subset of the plurality of computational nodes organized into a plurality of

processing layers having full lateral and feedback connections between the plurality of processing layers, at least one of the second subset of the plurality of computational nodes operatively being connected at least one of the first subset of the plurality of computational nodes;

a third subset of the plurality of computational nodes organized into at least one output channel operatively connected to a data storage device or network, at least one of the third subset of the plurality of computational nodes operatively being operatively connected to at least one of the second subset of the plurality of computational nodes comprising at least one re-entrant feedback channel,

wherein the feedforward input channel receives data from the external data source and transforms the data to a digital format and distributes the data in the digital format to at least one of second subset of the plurality of computational nodes,

wherein the data in the digital format is processed by the plurality of processing layers using the local update processes of the nodes comprising plurality of processing layers, using the full lateral and feedback connections within the processing layers and using re-entrant feedback from the re-entrant feedback channel such that the data in the digital format is combined and minimalized, and

wherein the combined data is output by the at least one output channel to the data storage device or network.